

# **Redes de Computadores**



**Cruzeiro do Sul Virtual**  
Educação a Distância



# Material Teórico



TCP/IP e roteamento

**Responsável pelo Conteúdo:**

Prof. Esp. Hugo Fernandes

**Revisão Textual:**

Prof. Ms. Luciano Vieira Francisco



# UNIDADE

## TCP/IP e roteamento



- Roteamento TCP/IP
- Roteamento Dinâmico
- Protocolos da Suíte TCP/IP
- Camada de Aplicação
- Camada de Transporte
- Camada de Rede



### OBJETIVO DE APRENDIZADO

- Estudar as técnicas usadas para encaminhar pacotes pela rede.
- Conhecer os principais protocolos usados na troca de informações, importantes para programação em Java em aplicativos que trafegam pela rede.
- Conhecer a característica de protocolo de roteamento e outros protocolos da suíte TCP/IP.
- Entender os mecanismos usados por protocolos para trafegar na rede de computadores.





# Orientações de estudo

Para que o conteúdo desta Disciplina seja bem aproveitado e haja uma maior aplicabilidade na sua formação acadêmica e atuação profissional, siga algumas recomendações básicas:



## Assim:

- ✓ Organize seus estudos de maneira que passem a fazer parte da sua rotina. Por exemplo, você poderá determinar um dia e horário fixos como o seu “momento do estudo”.
- ✓ Procure se alimentar e se hidratar quando for estudar, lembre-se de que uma alimentação saudável pode proporcionar melhor aproveitamento do estudo.
- ✓ No material de cada Unidade, há leituras indicadas. Entre elas: artigos científicos, livros, vídeos e sites para aprofundar os conhecimentos adquiridos ao longo da Unidade. Além disso, você também encontrará sugestões de conteúdo extra no item **Material Complementar**, que ampliarão sua interpretação e auxiliarão no pleno entendimento dos temas abordados.
- ✓ Após o contato com o conteúdo proposto, participe dos debates mediados em fóruns de discussão, pois irão auxiliar a verificar o quanto você absorveu de conhecimento, além de propiciar o contato com seus colegas e tutores, o que se apresenta como rico espaço de troca de ideias e aprendizagem.

# Roteamento TCP/IP

A finalidade do roteamento é escolher o melhor caminho que um pacote deve seguir para chegar ao seu destino. Os equipamentos que executam roteamento armazenam uma tabela na memória RAM, na qual há correspondência entre o endereço de destino e a interface que oferecer melhor encaminhamento.

Para que o roteamento seja possível, há alguns protocolos responsáveis por construir e manter as tabelas de roteamento. Esses protocolos baseiam-se em um algoritmo específico para dar melhor desempenho ao armazenamento e gerência das tabelas. Têm também como responsabilidade a troca de informações sobre os caminhos conhecidos para outros equipamentos. Dessa forma, os equipamentos que executam roteamento na rede conseguem montar tabelas mais consistentes para direcionar os pacotes pelas interfaces corretas.

Existem, basicamente, dois tipos de algoritmos de roteamento utilizados na arquitetura TCP/IP: vetor de distância – *distance-vector* – e estado do enlace – *link-state*.



Leia o texto de Alex Soares de Moura, intitulado Roteamento: o que é importante saber e disponível em: <https://goo.gl/jqXltW>.

## Roteamento Por Vetor de Distância

Neste tipo de roteamento, os equipamentos responsáveis por rotear possuem uma tabela com a melhor distância conhecida para os vários destinos alcançáveis e a interface de saída a ser usada para chegar até o destino. A tabela é atualizada, de tempos em tempos, pelos equipamentos diretamente conectados para mantê-la atualizada.

O algoritmo usado nesse tipo de roteamento segue o modelo do melhor caminho, desenvolvido por Bellman-Ford. Esse modelo serviu para rotear pacotes no início da internet e foi chamado de RIP. A Figura 1 apresenta o algoritmo de Bellman-Ford – os passos para se chegar ao melhor caminho estão exemplificados nesta Figura.

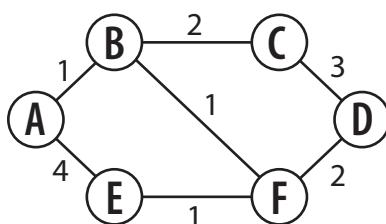


Figura 1 – Algoritmo Bellman-Ford  
Fonte: Gallo, 2003

O algoritmo de definição de rotas de Bellman-Ford é baseado em vetor de distância e itera no número de pulos entre um nó de origem e um de destino. Para ilustrar esse algoritmo, considere o seguinte gráfico não direcionado que ilustra uma rede. Os vértices A, B, C, D, E e F podem ser entendidos como roteadores e os arcos conectando esses vértices são canais de comunicação. Os rótulos dos arcos representam um custo arbitrário.

Nossa objetivo é encontrar o caminho mínimo de A a D usando o número de pulos como base para nossa seleção de caminho.

Examinamos os custos de todos os caminhos de A para cada um de nós com base no número de pulos.

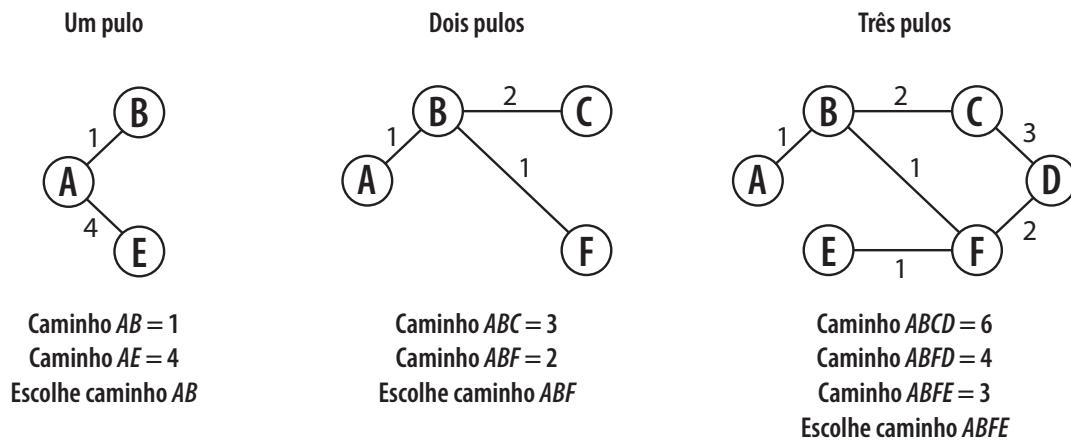


Figura 2  
 Fonte: Gallo, 2003

No último passo – três pulos –, dois caminhos são selecionados. O primeiro caminho, ABFD, representa o de custo mínimo de A a D com base na métrica de pulos. O segundo caminho, ABFE, é selecionado, pois representa o de custo mínimo de A a E.

O resultado final do algoritmo de Bellman-Ford é uma árvore representando o custo mínimo pago pelo nó de origem para todos os outros nós da rede. Árvores similares podem ser geradas para cada nó da rede. A árvore de custo mínimo do nó A em nosso exemplo é:

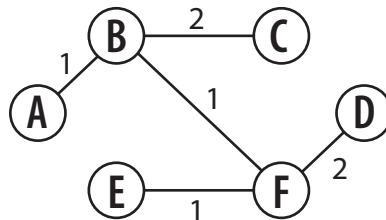


Figura 3  
 Fonte: Gallo, 2003

Assim, do nó A o caminho de custo mínimo para:

- B é AB = 1;
- C é ABC = 3;
- D é ABFD = 4;
- E é ABFE = 3;
- F é ABF = 2.

O princípio básico desse algoritmo está em determinar a distância entre a origem e o destino, calculando o número de saltos de roteadores necessários para um pacote chegar da rede de origem à rede de destino. Esse protocolo suporta, no máximo, quinze saltos, portanto, se um pacote tiver que alcançar uma rede que ultrapasse esses quinze saltos, não chegará, ou seja, no décimo sexto salto será descartado. A essa característica chamamos de rede “inalcançável” e isso nos faz concluir que esse tipo de protocolo de roteamento só nos garante quinze redes conectadas consecutivamente. Para manter as tabelas atualizadas, os equipamentos vizinhos trocam informações sobre a tabela de roteamento a cada trinta segundos. Isso não é interessante se tivermos uma rede de grandes proporções, pois uma boa parte da banda das conexões WAN será consumida com essas tabelas sendo trocadas entre os roteadores.

## Roteamento de Estado de Link

---

Este tipo de protocolo não envia toda a tabela de roteamento para os equipamentos vizinhos a cada atualização da tabela. Em vez disso, envia apenas informações sobre um determinado evento ocorrido na rede, o qual pode ser a perda ou a inserção de um link. Essas informações são enviadas por Anúncio de Estado de Link (LSA). Esse protocolo usa o mesmo conceito da estrutura de dados, conhecida como árvores, para buscar determinada informação sobre o endereço da rede e sobre a interface de saída. O algoritmo no qual esse protocolo se baseia chama-se algoritmo do caminho mínimo primeiro e foi desenvolvido por Dijkstra, que faz referência sobre a velocidade do caminho para montar a tabela de roteamento. A ideia básica do algoritmo de estado de enlace é a seguinte: cada equipamento responsável pelo roteamento deve:

- 1º Descobrir seus vizinhos e aprender seus endereços de rede;
- 2º Medir o retardo para cada um dos vizinhos;
- 3º Criar um pacote que informe tudo o que acaba de ser aprendido. Cada roteador constrói um pacote chamado Link State Packet (LSP), que contém seu nome, o nome de seus vizinhos e o custo necessário para chegar até esse;
- 4º Enviar esse pacote a todos os outros roteadores;
- 5º Calcular o caminho mais curto para cada um dos roteadores.

# Roteamento Dinâmico

Para que os pacotes trafeguem pela internet, é necessário que tenham informações referentes ao endereço IP de origem e destino. Para tomar a decisão sobre qual saída é a melhor para que o pacote chegue ao destino, o roteador deve armazenar um conjunto de informações que permita tomar tal decisão. Essas informações são organizadas em forma de tabela, que é chamada de tabela de roteamento. As tabelas de roteamento podem ser obtidas pelo roteador de duas formas: através do roteamento estático ou de um roteamento dinâmico. O roteamento estático é aquele em que as informações da tabela de roteamento são definidas pelo administrador de redes, ou seja, todas as informações referentes à saída na qual um pacote deve trafegar são definidas e configuradas pelo administrador de rede. Diferentemente, o roteamento dinâmico é definido por protocolos de roteamento, ou seja, basta que o administrador de rede configure um protocolo de roteamento para que a tabela de roteamento seja implementada automaticamente pelos algoritmos determinados nesses protocolos de roteamento.

Os protocolos de roteamento fazem uso de alguns algoritmos de roteamento para calcular o caminho de custo mínimo entre a origem e o destino. Os algoritmos de roteamento usam uma métrica de custo mínimo para determinar o melhor caminho. Alguns protocolos de roteamento usam métricas comuns, como a quantidade de saltos, ou seja, de roteadores visitados por um pacote a caminho de seu destino. Os algoritmos podem usar também atraso de propagação, largura de banda, tempo, utilização do canal, bem como métricas não comuns, como a taxa de erros.

As tabelas de roteamento são implementadas pelos roteadores através de informações trocadas entre os roteadores vizinhos. Quando configurado um protocolo de roteamento dinâmico, um algoritmo é executado por trás desse protocolo para informar quais são as redes que devem fazer parte da tabela de roteamento. Abaixo serão apresentados alguns protocolos de roteamento dinâmico.

## RIP

O Protocolo de Informações sobre Rotas (RIP) foi um dos primeiros protocolos de roteamento dinâmico. Usa um algoritmo de vetor de distância que determina a melhor rota através de uma métrica de pulos. É um protocolo eficiente quando usado em pequenas redes, pois este foi o objetivo quando da criação desse protocolo. Nessa época, não se imaginava que a internet teria um crescimento significativo, como ocorre atualmente. O RIP mantém as tabelas de rotas de uma rede atualizadas, transmitindo mensagens de atualização de tabelas a cada trinta segundos. Grande parte dos roteadores permite a configuração desse período de tempo. Após um dispositivo baseado em RIP receber uma atualização, compara-a com suas informações anteriores.

RIP é um protocolo que consome recursos como banda do link WAN para trocar informações das tabelas de roteamento. Além de consumir tempo do processador para definir o melhor caminho, deve ser inserido na tabela de roteamento para que um pacote chegue ao seu destino. A primeira versão do protocolo RIP não dá a possibilidade de trabalhar com técnicas do tipo VLSM e CIDR, pois, quando esse protocolo foi desenvolvido, não existiam os problemas encontrados atualmente na internet. Outra versão foi desenvolvida para corrigir esse problema, a qual possibilita trabalhar com a técnica VLSM e traz algumas correções de problemas encontrados na primeira versão.



OSPF é um protocolo que usa um algoritmo de estado de ligações; é especificamente projetado para redes IP grandes e heterogêneas. Usa, como métrica para estabelecer as rotas em sua tabela de roteamento, a carga de tráfego, atrasos de propagação, velocidade na linha e largura de banda, diferentemente do protocolo RIP – este que usa saltos, apenas. As atualizações feitas nesse tipo de protocolo não ocorrem em um período de tempo pré-estabelecido. Na verdade, as atualizações ocorrem em dois momentos: quando o roteador é configurado com um protocolo OSPF e somente quando ocorrer algum evento em que determinada rede fique indisponível. Além disso, não difunde tabelas completas de rotas para atualizar os roteadores vizinhos. Em vez disso, pequenos pacotes de estado de ligação, denominados anúncios de estado de ligações, contendo informações específicas sobre as ligações de redes de um roteador específico são transmitidos, ou seja, a quantidade de informações trocadas entre roteadores após ocorrer um evento é muito pequena, pois somente será relatado o evento ocorrido e não será transmitida toda a tabela – como é feito no protocolo RIP.

## IGRP

Este protocolo é proprietário da Cisco. Foi projetado com o objetivo de trazer melhorias para o protocolo RIP. Conforme a rede foi crescendo, o protocolo RIP passou a ficar limitado, de modo que o protocolo IGRP solucionou alguns problemas relacionados a rotas em redes grandes e heterogêneas. A principal diferença entre o protocolo IGRP e RIP é a métrica das rotas. O primeiro usa uma fórmula matemática que considera fatores como a largura de banda e atrasos para calcular o valor métrico; o segundo, conforme descrito, usa como métrica os saltos. Desse cálculo, a menor métrica é determinada com um caminho de menor custo e é este que deve compor a tabela de roteamento.

## EIGRP

É outro protocolo desenvolvido pela Cisco. É considerado um protocolo híbrido combinando as melhores características dos protocolos para a definição de rotas com base em vetores de distância e em estado de ligação. Por exemplo, usa mensagens de notificação para obter informações sobre roteadores vizinhos. Utiliza também um protocolo especialmente projetado, o protocolo de transporte confiável para transmitir as atualizações sobre rotas. As métricas de rotas são baseadas em vetores de distância e calculadas usando o algoritmo de difusão de atualização da Cisco (Dual).

# Protocolos da Suíte TCP/IP

Para que possamos realizar análise do comportamento de uma rede de computadores, é necessário entender como funcionam os principais protocolos de cada camada da suíte que é utilizada. Para nosso caso, usaremos a suíte TCP/IP, pois é a mais comum e utilizada atualmente para transmissão de dados pela internet. A Figura abaixo mostra alguns protocolos capturados por um analisador de protocolos.

Analizando essa Figura, podemos verificar que aparecem os protocolos HTTP, TCP, ARP e NBNS. Esses protocolos estão contidos na suíte TCP/IP e cada um tem um papel fundamental na troca de mensagens entre as máquinas. A análise do desempenho de uma rede, entre outros fatores, passa pelo exame das informações contidas nos pacotes que trafegam pela qual. Os administradores de redes devem usar ferramentas apropriadas para capturar esses pacotes a fim de analisá-los em momentos em que a rede não esteja mais surpreendendo as necessidades dos usuários. Sabendo dos pacotes que estão trafegando pela rede, esses administradores podem direcionar as decisões para sanar os problemas que ocorrem. Você estudará os principais protocolos da suíte TCP/IP. Isso lhe dará condições para avaliá-los através de uma ferramenta de análise de protocolo gratuita que poderá baixar da internet.

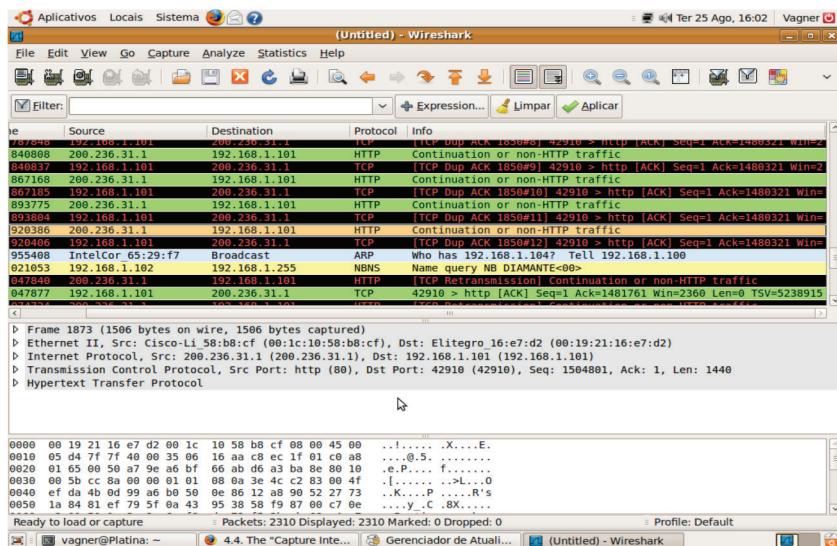


Figura 4 – Analisador de protocolo

Começaremos nosso estudo indo da camada superior para a inferior. Considerando a suíte TCP/IP, passaremos pelas camadas de aplicação, transporte e rede. Iniciaremos a análise pelos protocolos da camada de aplicação.

# Camada de Aplicação

Abaixo você encontrará os principais protocolos da camada de aplicação e suas características.

## Simple Mail Transfer Protocol (SMTP)

O SMTP corresponde a um dos serviços prestados pela camada de aplicação. Os componentes básicos envolvidos na troca de mensagens através do SMTP são:

- **Agente usuário** – corresponde a uma aplicação, ou seja, um programa para envio e recepção de mensagens. Um exemplo de agente usuário conhecido é o Outlook;
- **Caixa postal** – corresponde a um sistema de arquivos específicos de host, em que as mensagens destinadas a um determinado usuário ficam armazenadas até a sua recuperação. As caixas postais são configuradas nos servidores de e-mail.

O SMTP utiliza os serviços da camada de transporte através do protocolo TCP para transferir mensagens. Apresenta uma estrutura organizada para a transmissão dos dados. A Figura abaixo apresenta essa arquitetura:

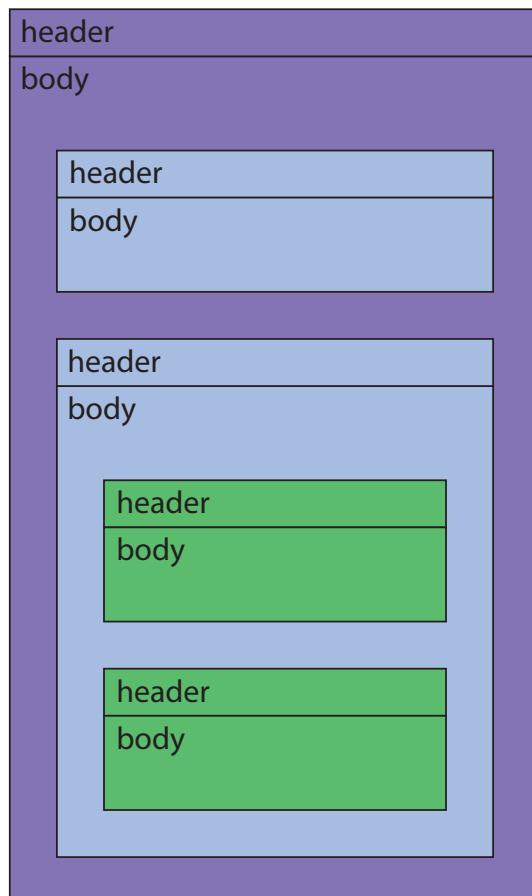


Figura 5 – Estrutura do protocolo SMTP

Como bem sabemos, o formato dos endereços definidos para o uso de e-mails é constituído de um campo de usuário, o qual identifica uma determinada conta dentro de um processo servidor, e um campo de domínio, o qual especifica uma organização. O caractere @ foi estabelecido como separador de campos. A Figura abaixo demonstra um mecanismo de troca de mensagens entre agentes usuários.

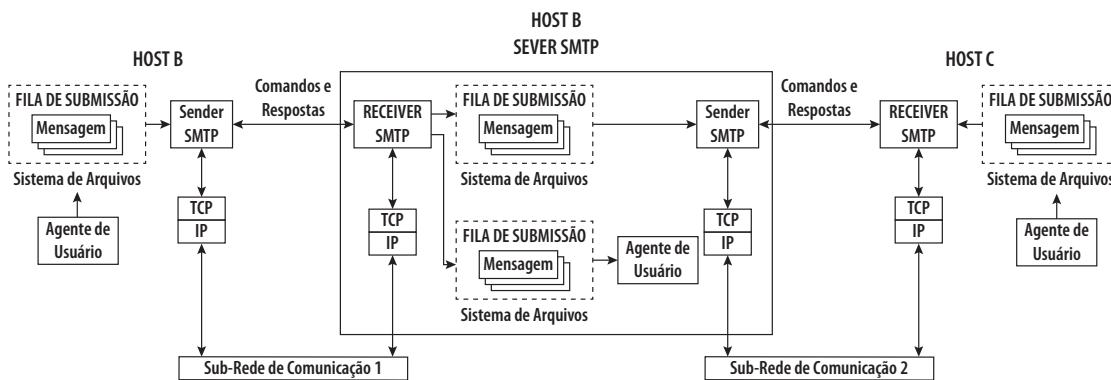


Figura 6 – Comunicação SMTP

## Telnet

É um protocolo para acesso remoto utilizado para configurações de máquinas. Fornece um serviço de terminal virtual. Uma vez estabelecida uma sessão de *login* remoto, *Telnet* disponibiliza mecanismos necessários para que os caracteres digitados na máquina local sejam passados diretamente à máquina remota. Não há interface gráfica; tudo é feito na linha de comando. Usa o protocolo TCP para o transporte confiável dos dados e é acessível através de programas de aplicação, chamados, na maioria das vezes, *Telnet*. Com este tipo de protocolo, os administradores de rede podem acessar, de forma remota, os roteadores ou outros dispositivos de rede que estão geograficamente distantes e implantados sobre o seu domínio.

## FTP

É um protocolo de transferência de arquivos entre dois sistemas que utiliza duas conexões: uma é empregada para dar suporte ao processo de transferência de dados e a outra conexão é utilizada para dar suporte aos vários processos de controle da sessão.

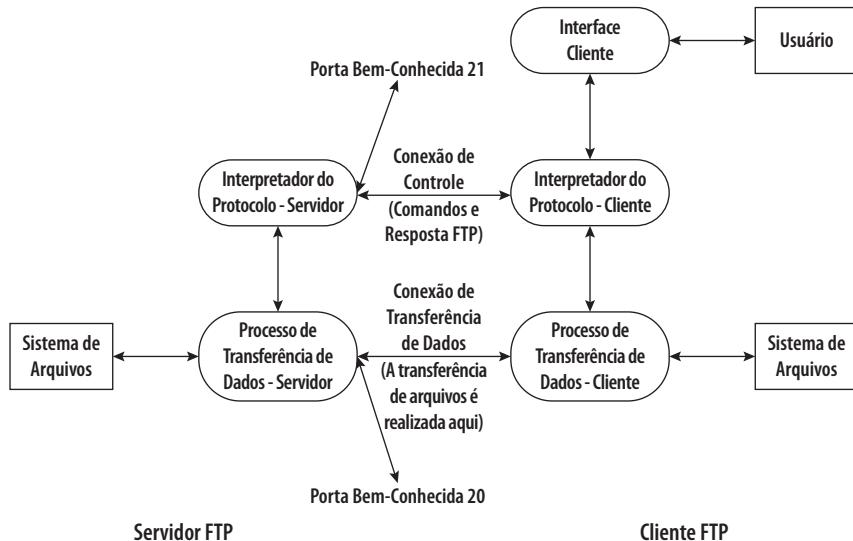


Figura 7 – Modelo conceitual do processo

Fonte: Gallo, 2003

Um servidor FTP recebe, na porta 21, um pedido inicial de conexão de um cliente FTP e, uma vez aceito e estabelecida a conexão, o processo de controle cria uma conexão TCP separada para transferência de dados usando a porta número 20. Há várias aplicações FTP disponíveis para baixar. Tais aplicações encapsulam a complexidade dos comandos que devem ser dados para a troca de arquivos.

## HTTP

O protocolo de transferência e hipertexto é o protocolo em que a *world wide web* está baseada. É considerado um protocolo de pedido e resposta e funciona da seguinte forma: um programa cliente estabelece uma conexão TCP com um programa servidor HTTP. O programa servidor aceita essa conexão e responde ao pedido do cliente. As mensagens de pedido são feitas por meio de um agente usuário, que conhecemos como *browser*. As mensagens de respostas são fornecidas pelo servidor após receber e interpretar a mensagem de pedido. As mensagens têm uma linha de início, um campo cabeçalho, uma linha em branco que significa o final do cabeçalho e um corpo de mensagem que contém o retorno da solicitação.

O HTTP, basicamente, utiliza dois métodos de conexão. Um dos quais, o mais básico, é aquele que envolve uma conexão simples entre cliente e servidor, possibilitando ao *browser* fazer requisições diretamente para o servidor, e o servidor envia as respostas conforme essas solicitações. Como não há conexões intermediárias, presume-se que o recurso solicitado esteja no próprio servidor. O outro método é aquele que envolve a presença de dispositivos intermediários para que a conexão se estabeleça. Os dispositivos intermediários são: *proxy* e *gateway*.



Leia sobre as funcionalidades e protocolos da camada de aplicação disponível em:  
<https://goo.gl/b4u4SY>

# Camada de Transporte

A camada de transporte tem dois protocolos principais na suíte TCP/IP: TCP e UDP.

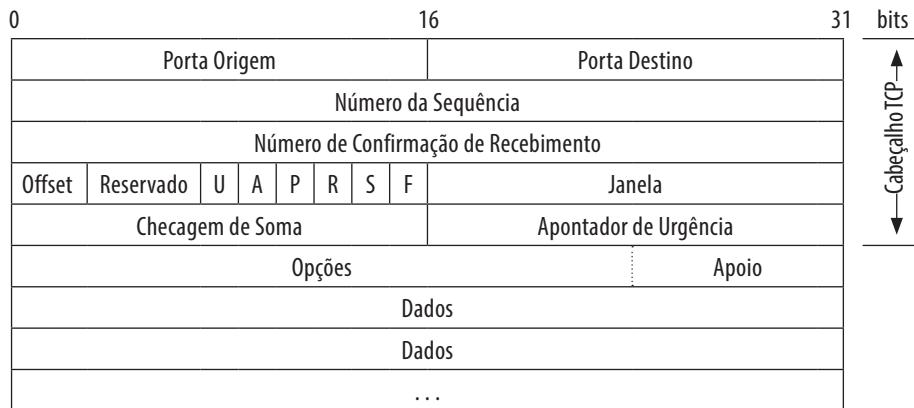


Figura 8 – Campos do protocolo TCP

Fonte: Gallo, 2003

O protocolo TCP é considerado confiável, ou seja, os mecanismos de entrega e o reconhecimento de recebimento de pacotes tornam-no um protocolo orientado à conexão e confiável. Significa que uma aplicação, trafegando dados pelo protocolo TCP, não precisa se preocupar com esses mecanismos que o tornam confiável. Alguns campos no cabeçalho são usados para essa finalidade. Entre os principais protocolos da camada de aplicação que usa o TCP, temos FTP, SMTP, POP. Na Figura 8 se encontram as descrições resumidas dos campos utilizados para protocolo TCP.

- **Porta de origem e porta de destino** – o TCP usa o número de porta para identificar e entregar dados para a aplicação correta;
- **Número de sequência** – este número é utilizado pela máquina receptora com o objetivo de montar de forma correta os segmentos recebidos. Os pacotes transmitidos são segmentados de acordo com a tecnologia utilizada; tais pacotes podem seguir por caminhos diferentes na internet e chegar ao destino de forma desordenada;
- **Número de confirmação de recebimento** – este campo é utilizado pela máquina transmissora com o objetivo de verificar se a máquina receptora recebeu o segmento. Se a máquina receptora não enviar esse número de confirmação para a máquina transmissora, considerando um determinado tempo, a máquina transmissora enviará novamente essa informação para a máquina receptora;
- **Controles** – são usados para controlar o envio e o recebimento dos pacotes pela internet;
- **Offset – 4 bits** – especifica o tamanho do cabeçalho do segmento TCP;
- **Campo reservado – 6 bits** – é reservado para uso futuro;

- **Flag URG** – indica que o seguimento tem dados urgentes, portanto, deve ser tratado com urgência pelos equipamentos intermediários;
- **Flag ACK** – indica que o segmento em questão tem dados de confirmação de recebimento;
- **Flag PSH** – indica que o segmento atual, o que é montado, tem dados e estes devem ser entregues imediatamente;
- **Flag RST** – é usada quando um evento causa uma desconexão indesejada. Quando isso ocorre, a máquina de origem envia um segmento TCP com esta flag setada para que a máquina de destino possa abortar o segmento;
- **Flag SYN** – indica que o segmento em questão contém dados no campo de número de sequência;
- **Flag FIN** – usada para terminar uma sessão TCP;
- **Campo janela – 2 bytes** – especifica o número máximo de informações que a máquina de destino é capaz de aceitar. Este campo é usado para controlar o fluxo de informações entre a origem e o destino. A quantidade de dados enviados ao destinatário não pode exceder a quantidade informada por este campo;
- **Checkagem de soma – este campo – 2 bytes** – é usado para verificar se o segmento transmitido é válido, ou seja, se houver algum erro na transmissão, este campo possibilitará que tal erro seja detectado;
- **Apontador de urgência – este campo – 2 bytes** – é usado em conjunto com a flag URG, que sinaliza para a máquina de destino que é necessário informar ao programa de aplicação que a informação enviada precisa ser processada com urgência;
- **Opções e apoio** – este campo tem o comprimento variável e especifica quais opções são requisitadas por um processo TCP. O tamanho máximo do segmento é uma opção geralmente usada. O campo apoio é empregado para preencher os bits restantes de um campo com zeros, a fim de que o pacote seja múltiplo de 32 bits.



Assista ao vídeo de Ailton Luiz Dias Siqueira Júnior sobre a camada de transporte, disponível em: <https://goo.gl/fE4zaE>.

## Estabelecimento de uma Conexão TCP

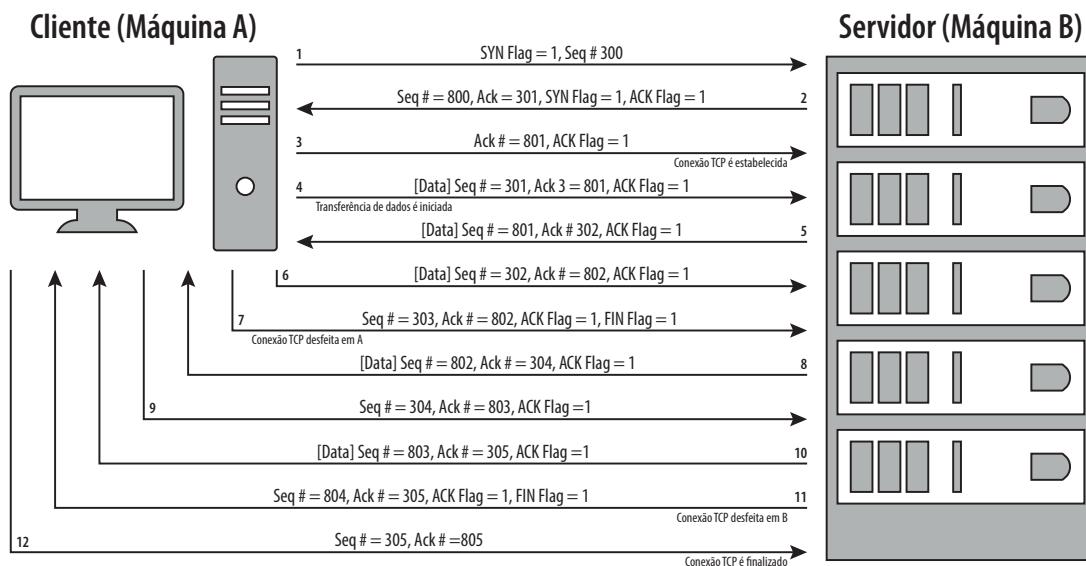


Figura 9 – Processo de comunicação TCP

Fonte: Gallo, 2003

1. A envia um segmento de sincronização a B, indicando seu desejo de estabelecer uma conexão e que seu número de sequência é 300. Significa que o primeiro segmento de dados que A envia será numerado 301;
2. B recebe o segmento de sincronização de A e envia um segmento de confirmação para A. Note que o número de sequência inicial de B é 800, significa que o primeiro segmento de dados de B será 801;
3. A recebe a sincronização e o segmento de confirmação de B e envia um segmento de confirmação. Neste estágio, uma conexão TCP é estabelecida entre A e B;
4. A transmite o segmento de dados 301 a B e informa a B que espera o número de sequência 801;
5. B recebe o segmento de A e envia o segmento de dados 801. Este segmento também confirma o recebimento do segmento de A, informando para A que espera receber o segmento 302;
6. A recebe o segmento de B e envia o segmento 302 a B, que confirma o recebimento do segmento de B;
7. A envia a B um segmento de finalização, que informa a B que A está rompendo o seu lado da conexão TCP;
8. B recebe os dois últimos segmentos de A e envia para A o segmento de dados 802. Note que este segmento também confirma o recebimento da transmissão anterior de A ajustando o número de confirmação 304. Neste ponto, A não pode transmitir nenhum novo segmento de dados, mas continua a transmitir os segmentos de confirmação;
9. A confirma a última transmissão de B;

10. *B* recebe o segmento de confirmação de *A* e envia o segmento de dados 803;
11. *B* envia para *A* o segmento de finalização, que informa para *A* que *B* está rompendo o seu lado da conexão TCP;
12. *A* recebe as últimas transmissões e confirmações de *B*. Neste ponto, o link é finalizado, uma vez que nem *A*, nem *B* têm mais dados a serem transmitidos.

O protocolo TCP é um protocolo orientado à conexão. Significa que alguns procedimentos de troca de mensagens devem ser estabelecidos antes da troca de dados.

Chamamos esse procedimento inicial de *three-way handshake* – comunicação em três fases – e envolve uma máquina cliente requisitando o estabelecimento de um *link* entre esse e o servidor. A Figura abaixo mostra a comunicação entre um cliente e um servidor e os campos trocados entre os quais para que se estabelecesse uma conexão.

## Protocolo UDP



Figura 10 – Campos do UDP

Este protocolo não é orientado à conexão, portanto, não tem campos necessários para estabelecer uma conexão confiável. Dessa forma, este tipo de protocolo não faz detecção ou correção de erros, não retransmite os dados que não foram recebidos e nem tem habilidade para lidar com erros ou controle de fluxo. Se este tipo de protocolo não garante a entrega do pacote, por que, então, usá-lo? Como pudemos perceber acima, o protocolo TCP tem mecanismos para entrega confiável das informações, no entanto, todo o mecanismo utilizado para criar essa confiabilidade acaba afetando a velocidade da entrega de pacotes.

Para as aplicações em que a velocidade é primordial para o bom desempenho, o protocolo UDP deve ser o escolhido. O único problema de utilizar o protocolo UDP é que a aplicação é que deve implantar os mecanismos para garantir que os pacotes sejam entregues, portanto, o controle de fluxo e todos os controles para identificar que os pacotes chegaram ao destino devem ficar a cargo da aplicação. Como pode ser observado na Figura abaixo, o datagrama UDP tem poucos campos e é por esse motivo que esse protocolo é processado mais rapidamente e, consequentemente, enviado e transmitido pela rede mais rapidamente.

Esses dois protocolos são os mais usados pelas aplicações. O uso de um ou outro depende da aplicação a ser empregada. Se alguma aplicação precisar enviar os dados de uma forma mais rápida, então o desenvolvedor da aplicação deverá optar pelo uso do UDP; agora, se a velocidade não for o problema, então o TCP poderá ser usado a fim de garantir os mecanismos de entrega dos pacotes.

Os campos estudados são importantes, pois a grande maioria dos analisadores de pacotes apresenta os conteúdos que estão trafegando nesses campos. Saber avaliar o que é transmitido ajuda na tomada de decisão em situações como: aumento do filtro para evitar que determinados sites sejam acessados, ou a troca de equipamentos para proporcionar maior velocidade na rede.

## Camada de Rede

O objetivo básico da camada de rede é fornecer os serviços de transferência de dados fim a fim sobre uma rede, independentemente das características das sub-redes físicas. Está mais relacionada à topologia de rede e tem como uma de suas principais funções resolver problemas de roteamento em rede.

A camada de rede deve executar as seguintes funções:

- **Roteamento** – as funções de roteamento determinam a rota apropriada entre endereços de redes;
- **Endereçamento dos usuários** – os serviços de rede utilizam um esquema de endereçamento que permite aos usuários referenciar, de maneira única, outros usuários;
- Fornecimento de serviços para a camada de transporte.

O principal protocolo dessa camada é o *Internet Protocol* (IP). É um datagrama, portanto, é livre de conexão e não garante a entrega das informações que nesse são encapsuladas. O IP recebe dados da camada de transporte, organiza esses dados como pacotes – datagrama IP – e seleciona a “melhor” rota com base nos critérios definidos na tabela de roteamento. Esses critérios são as métricas – qualidade da rota – definidas na configuração do roteamento. Por ser um datagrama, ou seja, não ser orientado à conexão, o IP deve levar, durante todo o percurso, o endereço IP de destino, pois cada datagrama relacionado a uma informação pode seguir caminhos diferentes na rede até chegar ao destino. O IP não tem mecanismos para tratar a perda de datagrama, portanto, se um datagrama for descartado por algum roteador, não será solicitada a sua retransmissão pela camada de rede; a responsabilidade para solicitar a retransmissão fica para as camadas superiores, neste caso, a camada de transporte.

O IP faz a segmentação dos dados para que não ultrapasse o que a tecnologia da camada inferior – de enlace – possa levar. A essa característica chamamos de fragmentação e à quantidade máxima de unidade de transmissão chamamos de *Maximum Transmission Unit* (MTU). A reestruturação dos pacotes é feita no equipamento de destino; não ocorre nos equipamentos intermediários pelos quais passam os pacotes.

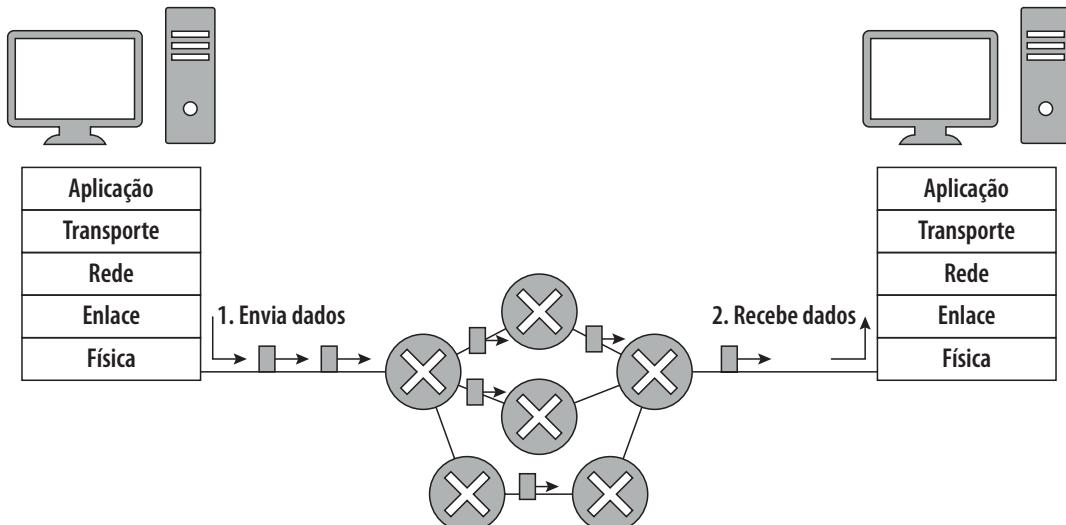


Figura 11 – Transmissão de pacotes

Fonte: Kurose, 2004

Assista ao vídeo sobre camada de rede em: <https://goo.gl/TsA3XD>.

Se tivermos muitos pacotes trafegando pela rede, o desempenho cairá, pois os roteadores podem ficar com muitos pacotes na fila – buffer – para serem analisados e podem, até, descartar alguns pacotes da fila caso falte espaço. A lentidão na entrega de pacotes ocorre, principalmente, em roteadores que são lentos no processamento dos dados. Em redes de computadores, tratamos essa lentidão como congestionamento.

Além do IP, que é considerado um protocolo roteável, a camada de rede tem outros protocolos, tais como ICMP, ARP e RARP. Nesta camada são definidos também os protocolos de roteamento, tais como RIP-v1, RIP-v2, IGRP, EIGRP, OSPF, IS-IS e BGP. Os protocolos roteáveis são aqueles que levam a informação do endereço de destino e os protocolos de roteamento são aqueles que usam algoritmos para construir tabelas de roteamento segundo métricas pré-estabelecidas.

## IP (IPv4)

IPv4 é o protocolo largamente usado para troca de dados atualmente. Por ser um protocolo de extrema utilização, detalharemos os campos que o compõem.

4	4	8	16	16	3	13	8	8	16	32	32	Variável	
V	HL	ST	TL	ID	F	F0	TTL	P	HC	SA	DA	OPT	PAD

Figura 12 – Campos do protocolo IPv4

Fonte: Gallo, 2003

Abaixo você conhecerá os detalhes de cada campo do protocolo IP:

- **Campo v** – contém 4 bits especificando a versão do protocolo. Se a versão for IPv4, este campo tem o valor 0100;

- **Campo HL** – tem 4 bits para identificar o tamanho do cabeçalho em 32 bits. Torna-se necessário porque os campos PAD e OPT são variáveis;
- **Campo ST** – com 8 bits, denominado tipo de serviço, este campo especifica como o pacote deve ser roteado. Como pode ser visto abaixo, tem três subcampos:

Precedência (3 bits)	Tipo de Serviço (4 bits)	MBZ (1 bit)
-------------------------	-----------------------------	----------------

Figura 13 – Formato do campo tipo de serviço  
 Fonte (Gallo, 2003)

- **Precedência** – especifica a prioridade do datagrama;
- **Tipo de serviço – TOS** – por este campo é que se especifica a qualidade de serviço em relação ao pacote, ou seja, informa se o pacote deve ter prioridade;
- **Campo MBZ** – não utilizado, deve ser preenchido com zero;
- **Campo TL** – especifica o tamanho do pacote. Este campo tem 16 bits, portanto, o máximo a que um pacote pode chegar é 65.535 bytes;
- **Campo ID** – usado para ajudar na remontagem dos pacotes fragmentados. Este campo tem 16 bits;
- **Campo FO** – de 3 bits, é responsável pela remontagem dos pacotes fragmentados;
- **Campo TTL** – contendo 8 bits, especifica o tempo de sobrevivência do pacote. Este campo é decrementado a cada roteador que o pacote passa; se chegar a zero, o pacote é descartado;
- **Campo P** – de 8 bits, contém informações do protocolo da camada 4;
- **Campo HC** – contém a checagem da soma referente ao cabeçalho. Dessa forma, mantém-se a integridade do cabeçalho;
- **Campo SA** – com 32 bits, é usado para especificar o endereço de origem;
- **Campo DA** – com 32 bits, é usado para especificar o endereço IP de destino;
- **Campo OPT** – reservado para opções de controle; tem tamanho variável;
- **Campo PAD** – usado em conjunto com o OPT para complementar o tamanho do cabeçalho em múltiplo de 32 bits. Complementa com zero a quantidade necessária para preencher os 32 bits restantes do pacote.

## Protocolo ICMP

---

O *Internet Control Message Protocol* (ICMP) é um protocolo da camada 3, usado para trocar mensagens entre equipamentos que estão interligados na rede. Em redes LAN, é empregado, pelos administradores de rede, para verificar se um determinado equipamento está devidamente conectado à rede. Para que isso seja possível, o comando *ping*, em conjunto com o endereço IP ou o nome do computador, é usado – portanto, por trás do comando ping, está o protocolo ICMP.

Se um destino recebe a solicitação de eco do ICMP, formula uma resposta de eco para enviar de volta à origem. Se o emissor recebe a resposta de eco, isso confirma que o destino pode ser alcançado.

O ICMP é usado também pelos roteadores para trocar mensagens de erros para o IP de forma automática. Quando há erros de entrega de um datagrama, o ICMP é utilizado para relatá-los ao emissor do datagrama. Esse não corrige o problema encontrado na rede, apenas relata ao emissor o status do pacote entregue, pois sua função não é propagar informações sobre alterações ocorridas na rede – como fazem os protocolos de roteamento dinâmico. As mensagens ICMP são encapsuladas em datagramas IP, aproveitando o endereçamento; no entanto, o ICMP tem seus próprios campos.

```
C:\> ping www.unicsul.br
C:\Users\Vagner>ping www.unicsul.br
Disparando www.unicsul.br [201.63.91.115] com 32 bytes de dados:
Resposta de 201.63.91.115: bytes=32 tempo=135ms TTL=121
Resposta de 201.63.91.115: bytes=32 tempo=114ms TTL=121
Resposta de 201.63.91.115: bytes=32 tempo=165ms TTL=121
Resposta de 201.63.91.115: bytes=32 tempo=110ms TTL=121

Estatísticas do Ping para 201.63.91.115:
  Pacotes: Enviados = 4, Recebidos = 4, Perdidos = 0 <0% de perda>
  Aproximar um número redondo de vezes em milissegundos:
    Mínimo = 110ms, Máximo = 165ms, Média = 131ms

C:\Users\Vagner>
```

Figura 14 – Comando ping bem-sucedido

Os formatos das mensagens do ICMP possuem três campos: tipo, código e checksum.

O campo tipo indica o tipo de mensagem do ICMP que é enviado. O campo código inclui informações adicionais específicas do tipo de mensagem e o campo checksum é utilizado para verificar a integridade dos dados.

Na Figura 15 é apresentado o modelo do pacote ICMP – encapsulado no pacote IP que, por sua vez, é inserido no quadro ethernet:

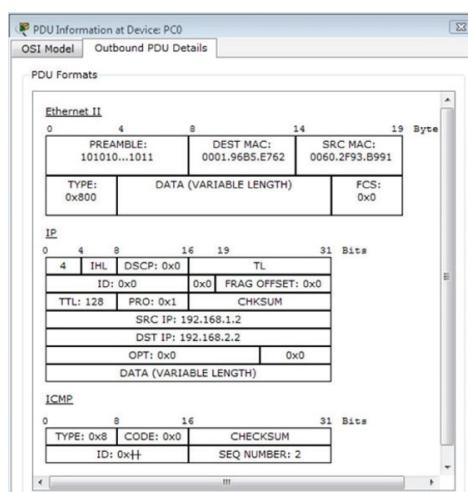


Figura 15 – Formato dos campos ICMP

# Material Complementar

## Indicações para saber mais sobre os assuntos abordados nesta Unidade:

### Sites

Visão geral do TCP/IP

<https://goo.gl/0E1jes>

### Vídeos

O QUE é gateway? Na prática – rede TCP-IP básico.

<https://youtu.be/wrHxvTcHE1k>

PROTOCOLO IP – endereçamento TCP-IP v4.0.

<https://youtu.be/EG9mSXIMTU4>

# Referências

DIÓGENES, Y. **Certificação Cisco:** CCNA4.0 – guia de certificação para o exame 640-801. Rio de Janeiro: Axcel Books do Brasil, 2004.

GALLO, M. A.; HANCOCK, W. M. **Comunicação entre computadores e tecnologias de rede.** São Paulo: Thomson Learning, 2003.

KUROSE, J. F. **Redes de computadores e a internet: uma nova abordagem.** São Paulo: Addison-Wesley, 2004.

SEMÉRIA, C. **Understanding IP addressing:** everything you ever wanted to know. [S.l.]: 3Com, 1996.



**Cruzeiro do Sul Virtual**  
Educação a Distância

www.cruzeirodosulvirtual.com.br  
Campus Liberdade  
Rua Galvão Bueno, 868  
CEP 01506-000  
São Paulo - SP - Brasil  
Tel: (55 11) 3385-3000



**Cruzeiro do Sul**  
Educacional